数理解析研究所講究録
625巻 1987年 157-166

ゲーム木探索法ＳＳＳ*の非劣性について

京都大学工学部　加藤　芳朗　(Yoshiroh KATOH)
京都大学工学部　茨木　俊秀　(Toshihide IBARAKI)

あらまし

完全情報２人零和ゲームはminimaxゲーム木で表現され、ゲーム木を解くことで両者が最適な手を選んだ場合の結果を知ることができる。この目的に、$\alpha-\beta$、ＳＳＳ*、双対ＳＳＳ*、Ｂ*、Ｈ*等の探索法が提案されている。

ゲーム木を探索しゲーム値を求めるアルゴリズムＡとＢにおいて、任意のゲーム木に対し、Ａの探索領域がＢの探索領域の部分集合であるとき、ＡはＢを凌ぐという。本稿では、ＳＳＳ*および双対ＳＳＳ*のそれぞれに対し、それを真に凌ぐ探索法は存在しないことを証明する。

１．まえがき

将棋、囲碁、オセロ等の完全情報２人零和ゲームは、minimaxゲーム木によって表現され、ゲーム木を解くことにより、両者が最適な手を選んだ場合の結果を知ることができる。ゲーム木を解くには、必ずしもゲーム木のすべての節点を探索しなくてもよいことから、必要な探索領域を減少させる試みが数多くなされ、$\alpha-\beta$ (5),(7)、ＳＳＳ*(8)、双対ＳＳＳ*(3)、Ｂ*(1)およびＨ*(2)などが知られている。

ゲーム木を探索しゲーム値を求めるアルゴリズムＡとＢにおいて、任意のゲーム木に対し、それを解くのに必要なＡの探索領域がＢの探索領域の部分集合であるとき、ＡはＢを凌ぐ(surpass)という。また、Ｂを凌ぐＡにおいて、あるゲーム木に対し、Ａの探索領域がＢの探索領域の真部分集合ならば、ＡはＢを真に凌ぐ(strictly surpass)という。

Stockman(8)は、1979年ＳＳＳ*法を提案し、ＳＳＳ*法が$\alpha-\beta$法を真に凌ぐことを示した。その後、$\alpha-\beta$法を凌ぐ探索法のクラスの完全な特徴付けが茨木(3)によって与えられ、さらに最近、茨木(4)によって、Ｂ*の変形である探索法Ｈ*に対し、Ｈ*を真に凌ぐ探索法は存在しないことが証明された。本稿では、ＳＳＳ*および双対ＳＳＳ*に対しても、そのような探索法が存在しないことを示す。

2. ゲーム木とその探索

2.1 ゲーム木

図1にminimaxゲーム木の例を示す。節点はゲームの各局面に対応し、各節点の子節点は、可能な次手を打つことによって得られる局面を表わす。端末節点(terminal node)は1つの終局を表わし、その局面の静的評価値が与えられている。ゲームは、MAXおよびMINの2人の競技者によって進められ、それぞれの手番に対応する局面を、MAX節点□とMIN節点○で示す。ゲーム木では、根から下へ向かう任意の路に沿って、MAX節点とMIN節点が交互に現れる。

各節点Pに対し、その局面から両プレイヤーが最善手を選び続けたとき、実現される終局の静的評価値をPのminimax値と呼び、f(P)と記す。ゲーム木の端末節点からはじめ、その他の節点を下から上へ次の計算にしたがって処理すれば、すべての節点Pのf(P)を計算することができる(図1参照)。ただし、端末節点のf値はその静的評価値に等しい。

$$f(P) := \begin{cases} \max\{f(Q) \mid Q \in S(P)\}, & P:\mathrm{MAX} \\ \min\{f(Q) \mid Q \in S(P)\}, & P:\mathrm{MIN}. \end{cases} \tag{2.1}$$

ここで、S(P)は節点Pの子節点の集合を表わす。ゲーム木の根節点をP_0とするとき、$f(P_0)$をゲーム値と呼び、$f(P_0)$を正確に求めることをゲームを解くという。図1の例では、$f(P_0)=45$である。ところで、ゲーム木が陽に与えられれば、式(2.1)によってゲーム値を計算できるが、ゲーム木は通常きわめて大規模であるため、式(2.1)の計算を直接実行することは困難である。そのため、ゲーム木の一部分

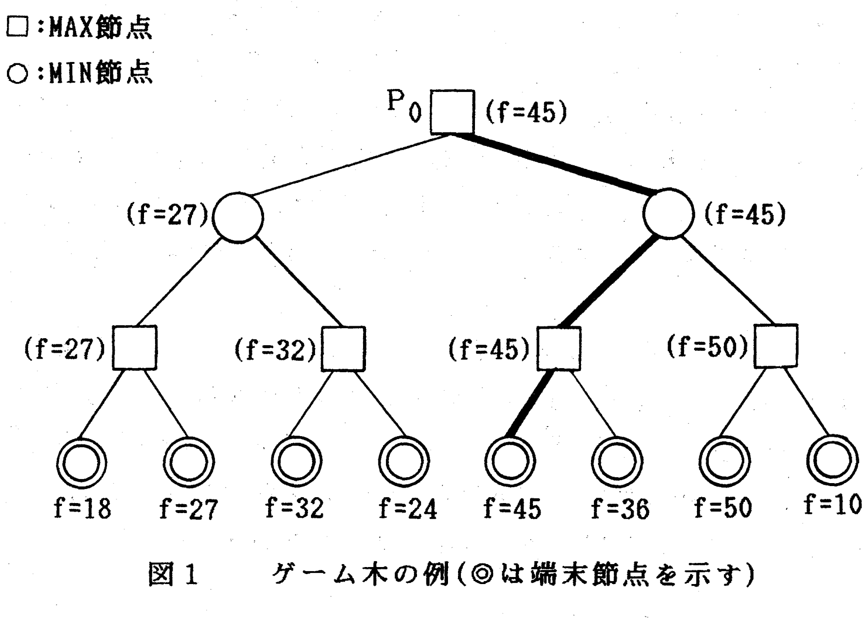

図1　ゲーム木の例(◎は端末節点を示す)

だけを実際に生成することでゲーム値を計算することを目的とした、種々の探索法が提案されている。

2．2　カット条件

ゲーム木の探索中において、ある節点Ｐの子孫の探索を進めてもゲーム値$f(P_0)$に影響を与えないことが明らかになれば、Ｐを根とする部分木の探索を考慮から除外することができる。このような操作をカット(cut)という。カットを効果的に利用することにより、探索時間の短縮が可能となる。各探索法に応じてカットを実際に適用する手順は異なってくるが、カットを可能にする条件は統一的に表現することができる。以下、文献[3]の記法に従って紹介するが、本稿では、同文献にあるような上下界値の利用は想定せず、端末節点に付された静的評価値のみを利用する、いわゆる静的モデルを対象としていることを断っておく。

与えられた探索法において、ある時点までに生成されたゲーム木の一部を探索木といい、Ｔと記す。Ｔはゲーム木の根P_0を含み、$P(\neq P_0)$がＴに含まれるならばその親も必ず含まれるという性質をもつ部分木である。ゲーム木自身も探索木になり得るが、一般の場合と区別するために、ゲーム木の全体をT_0と記すことがある。探索木Ｔの節点Ｐに対し、その子節点がＴに含まれていないならばＰは突節点(tip node)であるという。また、突節点でなければ中間節点という。T_0の端末節点は、Ｔに含まれているならば必ず突節点であるが、その逆は一般には成立しない。探索木Ｔの節点Ｐに対し、$f(P)$の上下界値$U_b(P)$および$L_b(P)$を、Ｔの下方から上方へ次の計算を適用することで求めることができる。

$$U_b(P)=\begin{cases} f(P), & P:\text{端末節点} \\ +\infty, & P:\text{Tの突節点であるがT0の端末節点でない} \\ \max\{U_b(Q) \mid Q\in S(P)\}, & P:\text{MAX中間節点} \\ \min\{U_b(Q) \mid Q\in S(P)\}, & P:\text{MIN中間節点}. \end{cases} \tag{2.2}$$

$$L_b(P)=\begin{cases} f(P), & P:\text{端末節点} \\ -\infty, & P:\text{Tの突節点であるがT0の端末節点でない} \\ \max\{L_b(Q) \mid Q\in S(P)\}, & P:\text{MAX中間節点} \\ \min\{L_b(Q) \mid Q\in S(P)\}, & P:\text{MIN中間節点}. \end{cases} \tag{2.3}$$

その名が示すとおり、

$$L_b(P)\leqq f(P)\leqq U_b(P) \tag{2.4}$$

の関係が一般的に成立する。次に、Ｐの先祖部分の情報を集約した$U_t(P)$、$L_t(P)$

を以下のように定義する。

$$U_t(P)=\min\{U_b(Q)\mid Q\in \mathrm{AMIN}(P)\} \tag{2.5}$$

$$L_t(P)=\max\{L_b(Q)\mid Q\in \mathrm{AMAX}(P)\}. \tag{2.6}$$

ただし、AMIN(P)(AMAX(P))は節点PのMIN先祖(MAX先祖)の集合を表わす。なお、本稿では、Pの先祖あるいは子孫というとき、P自身は含まれない。

以上の定義を用いて、節点Pのカットの必要十分条件は、次のように書かれる[(3)]。

$$\max[L_t(P),\ L_b(P)]\geqq\min[U_t(P),\ U_b(P)]. \tag{2.7}$$

すべての端末節点Pは、性質$L_b(P)=U_b(P)=f(P)$によって常にカット条件を満たす。また、探索木Tにおいて、まだ調べられていない突節点Pに対するカット条件は、次の予備カット条件に等しい。

$$L_t(P)\geqq U_t(P). \tag{2.8}$$

2．3　探索法

ゲーム木探索の一般的な手順は、次のように書かれる。まず、根節点P_0のみからなる探索木から計算を開始する。計算の各反復時点において、探索木の突節点の中でまだ調べられていない節点(このような節点をOPEN節点と呼ぶ)をリストOPENに保持しておき、その中の1つの節点Pを選択する。そのPが予備カット条件$L_t(P)\geqq U_t(P)$を満たさなければ、Pを探索する。すなわち、Pが端末節点ならば、静的評価値を計算し、そうでなければPを展開し(つまり、Pの子節点をすべて生成し)、探索木を成長させる。何回かの反復後、OPENリストが空になるまでに、$U_b(P_0)=L_b(P_0)(=f(P_0))$が成立し、計算が終了する。

<u>procedure</u>　GSEARCH(3)

G1(初期化): $\mathrm{OPEN}:=\{P_0\}$, $U_b(P_0):=\infty$, $L_b(P_0):=-\infty$。

G2(探索): $L_b(P_0)=U_b(P_0)$なら停止する。さもなければ、OPENから節点Pを1個選び、OPENから削除する。

G3(予備テスト): $L_t(P)\geqq U_t(P)$なら、G2に戻る。

G4(上下界値の計算): Pが端末節点ならば、f(P)を計算する。$L_b(P):=f(P)$, $U_b(P):=f(P)$とし、さらに現在の探索木のすべての節点のL_b, L_t, U_b, U_tの値を更新する。Pは必ずカット条件を満たすので、そのままG2に戻る。

G5(展開): Pの子節点Qをすべて生成し、仮に、$L_b(Q):=-\infty$, $U_b(Q):=\infty$と置いた後、すべてのQをOPENに加える。G2に戻る。　□

これまでに提案されたいろいろな探索法は、各反復において、OPENから取り出す節

点の選択法の相違として捉えることができる。例えば、代表的な探索法である$\alpha-\beta$法は、OPENの中で、最も左に位置する節点を選ぶ探索法であり、いわゆる深さ優先探索(depth-first search)に基づくものである。$\alpha-\beta$法の名の由来は、この場合、

$$\alpha(P)=\max_{Q\in \mathrm{AMAX\text{-}LS}(P)} L_b(Q)$$

$$\beta(P)=\min_{Q\in \mathrm{AMIN\text{-}LS}(P)} U_b(Q)$$

を用いると、予備カット条件(2.8)が$\alpha(P)\geqq\beta(P)$と書かれることによる。ただし、根節点P_0から節点Pに到る路を$\pi(P)$で示すとき、AMAX-LS(P)(AMIN-LS(P))は節点Pの先祖に位置するMAX節点(MIN節点)の子節点で、$\pi(P)$より左に位置する節点の集合である。

一方、SSS*、およびそれと双対的に定義される双対SSS*は、いわゆる最良下界探索(best-bound search)に基づく探索法であり、

$$\bar{\alpha}(P)=\max_{Q\in \mathrm{AMAX\text{-}LS}(P)} U_b(Q)$$

$$\bar{\beta}(P)=\min_{Q\in \mathrm{AMIN\text{-}LS}(P)} L_b(Q)$$

を用いて以下のように記述される。

SSS*: OPENの中で$\bar{\beta}(P)$を最大にする節点を選ぶ(同点の場合は左の節点を優先する)。

双対SSS*: OPENの中で$\bar{\alpha}(P)$を最大にする節点を選ぶ(同点の場合は左の節点を優先する)。

なお、SSS*は元の論文[8]では全く異なる形式で書かれていたが、文献[3]で上のように定式化できることが示された。双対SSS*は、文献[3]で導入された。

さて、ある節点Pに対して、ステップG4あるいはG5が適用されたとき(すなわち、Pが端末節点ならば$f(P)$が計算され、Pが中間節点ならばPの子節点が生成されたとき)、節点Pは探索されたといい、G5でPの子節点が生成されたとき、Pは展開されたという。このとき、任意のゲーム木に対して、

(a) 探索法Aは探索法Bが節点Pを探索するときに限ってPを探索する、

(b) 探索法Aは探索法Bが節点Pを展開するときに限ってPを展開する、

が成立するとき、探索法Aは探索法Bを凌ぐという。さらに、探索法Aが探索法Bを凌ぎ、かつAが探索(展開)しないある節点をBが探索(展開)するようなゲーム木が少なくとも1つ存在するとき、AはBを真に凌ぐという。探索法Aを真に凌ぐ探索法が存在しないとき、Aは非劣(unsurpassed)であると定義する。この議論に関して、以下の定理が知られている。

[定理2.1][8]　SSS*は$\alpha-\beta$法を真に凌ぐ。　□

また、最近、次の定理も証明された。

[定理２．２][(4)] H^*は非劣である。 □

本稿では、以下、ＳＳＳ*も非劣性を有することを証明するが、その前に、ＳＳＳ*の性質を述べる。

３．ＳＳＳ*の性質

ＳＳＳ*の性質を示すために、いくつかの定義を行う。ある探索木の節点Ｐに対し、任意の$Q \in$ AMIN-LS(P)がカット条件を満たすとき、Ｐは左MAX認証であるという。さらに、節点Ｐが、

$$\beta(P) > \alpha(P) \tag{3.1}$$

を満たすならば、節点Ｐは有資格(eligible)であるという(なおここでは、$\infty > \infty$、$-\infty > -\infty$と解釈する)。GSEARCHのＧ２で、OPENから常に有資格な節点だけを選ぶ探索法を有資格探索といい、以下の性質が知られている。

[補題３．１][(3)] 任意の有資格探索は$\alpha-\beta$法を凌ぐ。逆に、有資格探索ではない探索法はどれも$\alpha-\beta$法を凌がない。 □

さらに、ＳＳＳ*に関して、次の補題が知られている。

[補題３．２][(3)] ＴをＳＳＳ*によって生成された探索木とし、節点ＰをＳＳＳ*によって選ばれたOPEN節点とする。このとき、$U_t(P) > L_t(P)$が成り立てば(つまり、予備カット条件が成立しなければ)、Ｐは有資格かつ左MAX認証である。 □

この補題は、ＳＳＳ*が有資格探索の特別な場合であることを意味している。したがって、Stockmanによる定理2.1は、補題3.1と補題3.2からも直接得られる。

ここで、さらにいくつかの関数を定義する。ただし、AMAX-RS(P)はＰの先祖に位置するMAX節点の子節点で、$\pi(P)$の右に位置する節点の集合である。また、LB(P)はＰと同じ親を持ち、Ｐの左に位置する兄弟の集合、さらに、$\pi(P) = P_0, P_1, P_2, \cdots, P_n (= P)$とする。

$$f_{\text{AMIN-LS}}(P) = \min_{Q \in \text{AMIN-LS}(P)} f(Q)$$

$$f_{\text{AMAX-LS}}(P) = \max_{Q \in \text{AMAX-LS}(P)} f(Q)$$

$$f^k_{\text{AMAX-RS}}(P_n) = \max_{Q \in \text{AMAX-RS}(P_k)} f(Q), \quad 0 \leq k \leq n$$

$$\alpha^k_R(P_n) = \max_{Q \in \text{AMAX-RS}(P_k)} U_b(Q), \quad 0 \leq k \leq n$$

$$\beta_{LB}(P) = \min_{Q \in LB(P)} U_b(Q)$$

$$\bar{\beta}_{LB}(P) = \min_{Q \in LB(P)} L_b(Q).$$

これらの記法を用いて、節点PがＳＳＳ*によって探索、展開されるための必要十分条件が得られている。

［定理３．１］(6)　ＳＳＳ*によって生成された探索木のOPEN節点 P_n がＳＳＳ*によって探索される（P_n が非端末節点ならば、さらに展開される）ための必要十分条件は次の２条件が成立することである。

$$\text{(A)} \quad f_{AMIN\text{-}LS}(P_n) > f_{AMAX\text{-}LS}(P_n)$$

$$\text{(B)} \quad f_{AMIN\text{-}LS}(P_n) \geq f^{k'}_{AMAX\text{-}RS}(P_n).$$

ただし、$\pi(P_n) = P_0, P_1, \cdots, P_n$ とし、ｋ'は、

$$\min_{Q \in LB(P_k)} f(Q) = f_{AMIN\text{-}LS}(P_n)$$

を満たす最小のｋである（P_k はMAX節点）。　□

条件(A)，(B)はｆ値を使っているので、実際の探索に利用することはできない。そこで、ｆ値の上下界値に基づく十分条件を考える。

［補題３．３］(9)　P_n 等を定理3.1のように定める。次の２つの条件、

$$\text{(A')} \quad \bar{\beta}(P_n) > \bar{\alpha}(P_n) \qquad \text{(有資格性条件)}$$

$$\text{(B')} \quad \bar{\beta}(P_n) \geq \bar{\alpha}^{k^*}_{R}(P_n)$$

が成立すれば条件(A),(B)が成り立つ。ただし、

$$\bar{\beta}_{LB}(P_k) < \beta(P_{k-2})$$

を満たす最大のｋ（$0<k\leq n$, P_k:MAX）をｋ*とする。　□

この十分条件に関して、次の補題が成立する。

［補題３．４］(9)　ＳＳＳ*によって生成された任意の探索木において（ただし、OPEN節点の存在を仮定する）、条件（Ａ'），（Ｂ'）を満足し、かつ左MAX認証であるOPEN節点 P_n は、$\bar{\beta}(P_n) = \infty$ である場合を除けば、ちょうど１個しか存在しない。　□

［補題３．５］(9)　任意の探索法Ａによって生成された探索木Ｔにおいて、条件（Ｂ'）を満足しない節点 P_n が存在すると仮定する。このとき、Ｔを拡大して得られる、あるゲーム木 T_0 において、ＳＳＳ*は P_n を探索しない。　□

４．非劣性定理

４．１　証明の方針

以下、ＳＳＳ*の非劣性を示すために、ＳＳＳ*と異なる任意の探索法Ａを想定し、ＡはＳＳＳ*を真に凌ぐことはないことを示す。そのために、ＳＳＳ*の計算が、その

選択規則から明らかなように、次の２段階に分かれることに注意しよう。

段階１：β(Ｐ)＝∞を満たす節点Ｐ∈OPENのみを探索する。

段階２：β(Ｐ)＜∞を満たす節点Ｐ∈OPENのみを探索する。

したがって、この性質と補題3.4を考慮すれば、上記のＡは次の５つの場合のどれかを引き起こす。

[Ⅰ]　Ａは有資格条件(すなわち条件(Ａ'))を満足しない節点Ｘを探索する。

[Ⅱ]　Ａは左MAX認証でない有資格節点Ｘを探索する。

[Ⅲ]　まだβ(Ｐ)＝∞を満たすOPEN節点Ｐが存在しているにもかかわらず、Ａはβ(Ｘ)≠∞かつ左MAX認証である有資格節点Ｘを探索する。

[Ⅳ]　段階１を経た後、段階２において、Ａは左MAX認証かつ有資格であるが条件(Ｂ')を満足しない節点Ｘを探索する。

[Ⅴ]　Ａは[Ⅰ]～[Ⅳ]のどの場合も引き起こさないが、β(Ｐ)＝∞を満たす節点の探索順序がＳＳＳ*と異なる。

場合[Ⅴ]においては、後で補題4.5で示すように、任意のゲーム木に対して、ＡとＳＳＳ*の探索領域は同一であり、したがって、ＡがＳＳＳ*を真に凌ぐことはない。[Ⅰ]については、ＡはＳＳＳ*を凌がないことを容易に示すことができ、本節で述べる。場合[Ⅱ]、[Ⅲ]、[Ⅳ]については、節点Ｘをそのような最初の節点と仮定し、節を分けて議論する。いずれの場合も、Ａが探索、展開するにもかかわらずＳＳＳ*は探索、展開を行わないような節点をもつゲーム木を構成し、ＡがＳＳＳ*を凌がないことを示す。

[補題４．１]　場合[Ⅰ]、すなわちＡが有資格でない節点を探索したとすると、ＡはＳＳＳ*を真に凌がない。

(証明)　定理2.1と補題3.1より、ＳＳＳ*を凌ぐ探索法が存在するとすれば、それは有資格探索でなければならないことから明らか。　□

４．２　[Ⅱ]の場合の証明

[補題４．２][9]　有資格探索である探索法Ａが、探索木Ｔ'において、初めて左MAX認証でない有資格節点Ｘを探索したと仮定する。このときＴ'を拡大して得られる、あるゲーム木T_0に対しては、その後Ａによって生成されたある探索木において、Ａは条件(Ｂ')を満たさないOPEN節点を探索する。(したがって、補題4.1と補題3.5によって、ＡはＳＳＳ*を凌がない。)　□

４．３　[Ⅲ]の場合の証明

[補題４．３][9]　探索法Ａは、探索木Ｔにおいて、まだβ(Ｐ)＝∞であるOPEN節

点Pが存在しているにもかかわらず、$\beta(X) \neq \infty$である左MAX認証かつ有資格な節点Xを初めて探索したとする。このとき、Tを拡大して得られる、あるゲーム木T_0において、ＳＳＳ*は節点Xを探索しない。 □

４．４ ［Ⅳ］の場合の証明

［補題４．４］(9) 段階１を経て段階２に移った後の探索木Tにおいて、探索法AとＳＳＳ*が初めて異なるOPEN節点XとYを探索したとする。このとき、Tを拡大して得られる、あるゲーム木T_0において、ＳＳＳ*は節点Xを探索しない。 □

４．５ ［Ｖ］の場合の証明

［補題４．５］(9) ［Ｖ］の場合、任意のゲーム木T_0に対して、AとＳＳＳ*の探索領域は同一である。 □

４．６ 非劣性定理

本章の結果をまとめ、次の定理を得る。

［定理４．１］ ＳＳＳ*は非劣である。 □

また、βとα等の双対性より、次の系の証明もＳＳＳ*の場合と並行的に行うことができる。

［系４．１］ 双対ＳＳＳ*は非劣である。 □

５． むすび

本論文では、最初に述べたように、ゲーム木としていわゆる静的モデル(static model)を仮定して議論を進めた。しかし、ＳＳＳ*の非劣性は、より一般的な情報付きモデル(informed model)(3)においても当然成立する(静的モデルは情報付きモデルの特別の場合になっているから)。

ところで、探索法Aが探索法Bを凌ぐという条件は非常に強いので、非劣であるような探索法は数多く存在するものと思われる。したがって、非劣性が実際の応用において、効率的な探索法であることをただちに意味するとはいえない。今後、非劣である探索法の中で、効率の良さをより精密に比較できるような他の尺度を導入する必要があろう。

謝辞

日頃ご討論いただく研究室の諸氏に感謝致します。なお、本研究は一部、文部省科学研究費によるものである。

文　献

(1) H. J. Berliner: "The B* tree search algorithm: A best-first proof procedure", Art. Int., 12, 23-40(1979).

(2) T. Ibaraki, S. Suzuki, K. Inoue and T. Hasegawa: "Heuristic search algorithm H* for solving minimax game trees", Kyoto Univ., Working Paper(1982).

(3) T. Ibaraki: "Generalization of alpha-beta and SSS* search procedures", Art. Int., 29, 73-117(1986).

(4) T. Ibaraki: "Game solving procedure H* is unsurpassed", to appear in Discrete Algorithms and Complexity Theory, edited by T. Nishizeki, Academic Press.

(5) D. E. Knuth and R. W. Moore: "An analysis of alpha-beta pruning", Art. Int., 6, 293-326(1975).

(6) I. Roizen and J. Pearl: "A minimax algorithm better than alpha-beta? Yes and no", Art. Int., 21, 199-220(1983).

(7) J. R. Slagle and J. K. Dixon: "Experiments with some programs that search game trees", J. ACM, 16, 189-207(1969).

(8) G. C. Stockman: "A minimax algorithm better than alpha-beta ?", Art. Int., 12, 179-196(1979).

(9) 加藤，茨木："ゲーム木探索法ＳＳＳ*の非劣性について"，信学技報，COMP86-69(1987-01).